

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problem Mailbox.**

(19)



JAPANESE PATENT OFFICE

PATENT ABSTRACTS OF JAPAN

(11) Publication number: **2000163107 A**

(43) Date of publication of application: **16 . 06 . 00**

(51) Int. Cl

**G05B 19/05**

(21) Application number: **10340596**

(22) Date of filing: **30 . 11 . 98**

(71) Applicant: **MITSUBISHI ELECTRIC CORP**

(72) Inventor:  
**WATABE HIROSHI  
KOIDE NAOMOTO  
YAMADA SHIGEKI  
OCHIAI KOGETSU  
OGUCHI SHUJI**

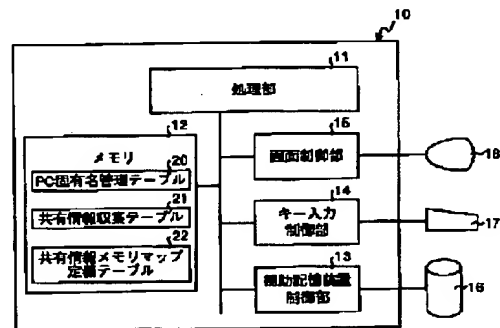
(54) **PROGRAM EDITING DEVICE FOR  
PROGRAMMABLE CONTROLLER**

(57) Abstract:

PROBLEM TO BE SOLVED: To efficiently and accurately prepare a control program to be used in a programmable controller with which plural programmable controllers are linked.

SOLUTION: A memory 12 is provided with a shared information collection table 21 for registering a programmable controller specific name for registering information for specifying a programmable controller to which a device is belonging and a device name being an interlock condition. Then, a trigger program including an outside contact and a shared information contact is automatically added to the sequence program of the programmable controller based on the data registered in the shared information collection table 21, and the shared information contact of the trigger program is automatically added to the sequence program of the programmable controller to be inter-PC interlocked as an inter-PC interlock contact so that program preparation processing can be attained.

COPYRIGHT: (C)2000,JPO



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2000-163107

(P2000-163107A)

(43) 公開日 平成12年6月16日 (2000.6.16)

(51) Int.Cl.<sup>7</sup>

G 0 5 B 19/05

識別記号

F I

G 0 5 B 19/05

テーマコード(参考)

B 5 H 2 2 0

審査請求 未請求 請求項の数 6 O L (全 20 頁)

(21) 出願番号 特願平10-340596

(22) 出願日 平成10年11月30日 (1998. 11. 30)

(71) 出願人 000006013

三菱電機株式会社

東京都千代田区丸の内二丁目2番3号

(72) 発明者 渡部 拓

東京都千代田区丸の内二丁目2番3号 三

菱電機株式会社内

(72) 発明者 小出 直基

愛知県名古屋市中区東大曾根町上五丁目

1071番地三菱電機メカトロニクスソフトウ

エア株式会社内

(74) 代理人 100089118

弁理士 酒井 宏明 (外1名)

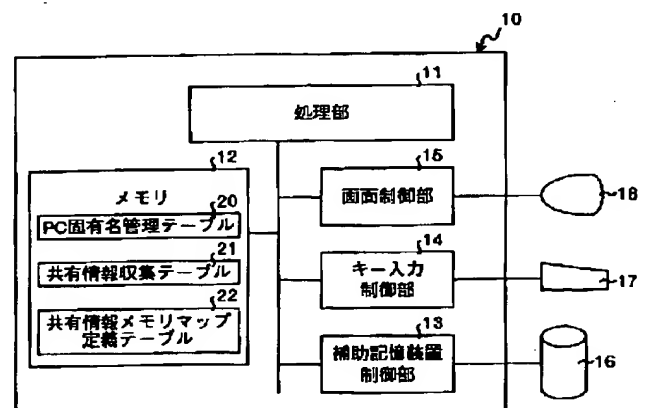
最終頁に続く

(54) 【発明の名称】 プログラマブルコントローラのプログラム編集装置

(57) 【要約】

【課題】 複数のプログラマブルコントローラが連携するプログラマブルコントローラにおいて使用する制御プログラムのプログラム作成を効率よく的確に行うこと。

【解決手段】 デバイスが属するプログラマブルコントローラを特定する情報を登録するプログラマブルコントローラ固有なと、インターロック条件となるデバイスの名前を登録する共有情報収集テーブル21をメモリに有し、共有情報収集テーブル21に登録されているデータに基づいて、該当するプログラマブルコントローラのシーケンスプログラムに外部接点と共有情報接点を含むトリガープログラムを自動的に付加し、P C間インターロック対象のプログラマブルコントローラのシーケンスプログラムに前記トリガープログラムの共有情報接点をP C間インターロック接点として自動的に付加するプログラム作成処理を行う。



**【特許請求の範囲】**

【請求項 1】 複数のプログラマブルコントローラが連携するプログラマブルコントローラで使用するシーケンスプログラムを作成するプログラム編集装置において、デバイスが属するプログラマブルコントローラを特定する情報を登録するプログラマブルコントローラ固有名と、インターロック条件となるデバイスの名前を登録する共有情報収集テーブルをメモリに有し、前記共有情報収集テーブルに登録されているデータに基づいて、該当するプログラマブルコントローラのシーケンスプログラムに外部接点と共有情報接点を含むトリガープログラムを自動的に付加し、PC間インターロック対象のプログラマブルコントローラのシーケンスプログラムに前記トリガープログラムの共有情報接点をPC間インターロック接点として自動的に付加するプログラム作成処理を行うことを特徴とするプログラマブルコントローラのプログラム編集装置。

【請求項 2】 PC間インターロック対象のプログラマブルコントローラの元シーケンスプログラムに設定されたPC間インターロック接点指示とPC間インターロック終了指示の間のすべてのプログラムに前記PC間インターロック接点を自動的に付加するプログラム作成処理を行うことを特徴とする請求項 1 に記載のプログラマブルコントローラのプログラム編集装置。

【請求項 3】 前記PC間インターロック接点指示はPC間インターロックのプログラマブルコントローラ固有名とインターロック条件となるデバイスの記述部を有し、前記元シーケンスプログラムにおけるPC間インターロック接点指示を抽出し、PC間インターロック接点指示のデータより前記共有情報収集テーブルを自動作成することを特徴とする請求項 2 に記載のプログラマブルコントローラのプログラム編集装置。

【請求項 4】 複数のプログラマブルコントローラが連携するプログラマブルコントローラで使用するシーケンスプログラムを作成するプログラム編集装置において、デバイスが属するプログラマブルコントローラを特定する情報を登録するプログラマブルコントローラ固有名と、インターロック条件となるデバイスの名前を登録する共有情報収集テーブルをメモリに有し、各ステップ毎に各プログラマブルコントローラの固有名を付けられ、トランジション条件としてPC間インターロック接点を記述された一つの順次実行型プログラムのステップに前記共有情報収集テーブルに登録されているデータに基づいて外部接点と共有情報接点を含むトリガープログラムを自動的に付加し、トランジション条件として該当するステップ内のシーケンスプログラムに前記トリガープログラムの共有情報接点をPC間インターロック接点として付加し、前記一つの順次実行型のプログラムを各ステップ毎に分離して各プログラマブルコントローラのシーケンスプログラムに自動的に変換するプロ

グラム作成処理を行うことを特徴とするプログラム編集装置。

【請求項 5】 各ステップ毎に付けられたプログラマブルコントローラの固有名とトランジション条件として一つの順次実行型のプログラムに記述されたPC間インターロック接点とから、前記共有情報収集テーブルを自動作成することを特徴とする請求項 4 に記載のプログラマブルコントローラのプログラム編集装置。

【請求項 6】 前記共有情報収集テーブルのデータ内容より各プログラマブルコントローラ毎の共有情報接点のデバイス範囲を自動検出し、各プログラマブルコントローラ毎の共有情報接点のデバイス範囲を示す共有情報メモリ定義テーブルを自動作成することを特徴とする請求項 1～5 のいずれか一つに記載のプログラマブルコントローラのプログラム編集装置。

**【発明の詳細な説明】****【0001】**

【発明の属する技術分野】 本発明は、プログラマブルコントローラで使用するシーケンスプログラムをプログラミングするプログラム編集装置に関し、特に、機械設備群を制御する複数のプログラマブルコントローラが連携するプログラマブルコントローラで使用するシーケンスプログラムのプログラミングを行うプログラム編集装置に関するものである。

**【0002】**

【従来の技術】 従来より、機械設備のシーケンス制御を行うために、プログラマブルコントローラ（以下「PC」という）が用いられており、複雑な機械設備を制御するために、複数のPCを相互に連携させてシステムを構築する場合がある。PC間での連携とは、自PC内情報のうち他PCと共有すべき情報を公開し、他PCが公開した共有情報を参照して自PCの制御の開始または停止を行うことである。

【0003】 図 23 は互いに連携する複数のPCによって機械設備のシーケンス制御を行うPCリンクシステムの一般例を示している。1 台目のPC本体（CPUユニット）50<sub>1</sub>、2 台目のPC本体50<sub>2</sub>～n 台目のPC50<sub>n</sub>のそれぞれに通信ユニット60<sub>1</sub>～60<sub>n</sub>が接続され、通信ユニット60<sub>1</sub>～60<sub>n</sub>がネットワーク通信ライン70によって相互に通信可能に接続されている。

【0004】 PC本体50<sub>1</sub>～50<sub>n</sub>のメモリ51<sub>1</sub>～51<sub>n</sub>には自機の制御に必要な情報を格納するPC内情報メモリ部52<sub>1</sub>～52<sub>n</sub>があり、このPC内情報メモリ部52<sub>1</sub>～52<sub>n</sub>の一部に他のPC（他機）に対して自機が公開する情報を格納する公開情報部分53<sub>1</sub>～53<sub>n</sub>が存在する。

【0005】 親機PC、ここでは、1 台目のPC本体（CPUユニット）50<sub>1</sub>の通信ユニット60<sub>1</sub>のメモリ61にはデータ共有情報メモリ部62が割り付けてある。データ共有情報メモリ部62のデータはPC間ネッ

トワーク通信によって受け渡され、接続されたPC間で共有される情報であり、データ共有情報メモリ部62は、1台目のPCの公開情報部分53<sub>1</sub>と同内容の公開情報部分62<sub>1</sub>、2台目のPCの公開情報部分53<sub>2</sub>と同内容のメモリ部62<sub>2</sub>～n台目のPCの公開情報部分53<sub>n</sub>と同内容の公開情報部分62<sub>n</sub>による各PCの公開情報部分から成り立っている。

【0006】PC本体50<sub>1</sub>内の公開情報部分53<sub>1</sub>とデータ共有情報メモリ部62内の公開情報部分62<sub>1</sub>、PC本体50<sub>2</sub>内の公開情報部分53<sub>2</sub>とデータ共有情報メモリ部62内の公開情報部分62<sub>2</sub>～PC本体50<sub>n</sub>内の公開情報部分53<sub>n</sub>とデータ共有情報メモリ部62内の公開情報部分62<sub>n</sub>は、データ共有情報メモリ部62のデータがネットワーク通信により受け渡される間に、それぞれのPCが自機の内部の公開情報部分の情報を、対応するデータ共有情報メモリ部62内の公開情報部分62<sub>1</sub>～62<sub>n</sub>に反映することで、常時同一内容に保たれている。

【0007】これにより各PCは、データ共有情報メモリ部62の共有情報を参照することにより他PCの状態を知ることが可能となる。各PCは他PCと連携してシーケンス制御を行うためのシーケンスプログラムに従って制御を行う。

【0008】図24、図25は、PC本体50<sub>1</sub>とPC本体50<sub>2</sub>とで連携して動作する場合のシーケンスプログラムを、一般的に普及しているラダー回路を用いて示したものである。

【0009】図24において、100はPC本体50<sub>1</sub>の外部接点(X1)の状態がオンであるかどうかを判定する条件判定命令であり、101は左記条件が真であれば、データ共有情報メモリ部62内の共有情報接点(B0)をオンする出力命令部である。

【0010】図25において、110はデータ共有情報メモリ部62内の共有情報接点(B0)がオンであるかどうかを判定する条件判定命令であり、111はPC本体50<sub>2</sub>の外部接点(X0)がオンであるかどうかを判定する条件判定命令であり、112は左記条件判定命令が直列に並んでいるので、それら条件がそれぞれ真の場合のみPC本体50<sub>2</sub>の内部接点(Y10)をオンする出力命令部である。

【0011】図24に示されているラダー回路と図25に示されているラダー回路が連携することにより、PC本体50<sub>1</sub>の外部接点(X1)100がオンし、さらにPC本体50<sub>2</sub>の外部接点(X0)111がオンした場合に、PC本体50<sub>2</sub>の内部接点(Y10)112がオンすることになる。

【0012】このように、PC間で連携する場合には、データ共有情報メモリ部62のデータを相互に出力、参照することができる。

【0013】

【発明が解決しようとする課題】上述した従来システムでは、共有可能な情報量は有限であるので、それぞれのPC間で共有する情報は、事前に連携するすべてのPC間で取り決めておくことが必要であり、このためシーケンスプログラムの作成が複雑化するという問題点がある。

【0014】また、従来システムでは、複数のPC間で連携するための、共有情報の範囲の設定を各PC毎に行う必要があり、このためシーケンスプログラムの作成が複雑化し、設定情報に誤りがあると、互いのPCが正常に動作しないという問題点がある。

【0015】また、従来システムでは、共有情報状態による条件判定命令を自PCの制御を行うためのシーケンスプログラムに付加しなければならず、シーケンスプログラムの作成が複雑化する問題点がある。

【0016】この発明は、上述の如き問題点を解消するためになされたもので、複数のプログラマブルコントローラが連携するプログラマブルコントローラで使用するシーケンスプログラムのプログラム作成を効率よく的確に行うプログラム編集装置を得ることを目的とする。

【0017】

【課題を解決するための手段】上述の目的を達成するために、この発明によるプログラマブルコントローラのプログラム編集装置は、複数のプログラマブルコントローラが連携するプログラマブルコントローラで使用するシーケンスプログラムを作成するプログラム編集装置において、デバイスが属するプログラマブルコントローラを特定する情報を登録するプログラマブルコントローラ固有名と、インターロック条件となるデバイスの名前を登録する共有情報収集テーブルをメモリに有し、前記共有情報収集テーブルに登録されているデータに基づいて、該当するプログラマブルコントローラのシーケンスプログラムに外部接点と共有情報接点を含むトリガープログラムを自動的に付加し、PC間インターロック対象のプログラマブルコントローラのシーケンスプログラムに前記トリガープログラムの共有情報接点をPC間インターロック接点として自動的に付加するプログラム作成処理を行うものである。

【0018】つぎの発明によるプログラマブルコントローラのプログラム編集装置は、PC間インターロック対象のプログラマブルコントローラの元シーケンスプログラムに設定されたPC間インターロック接点指示とPC間インターロック終了指示の間のすべてのプログラムに前記PC間インターロック接点を自動的に付加するプログラム作成処理を行うものである。

【0019】つぎの発明によるプログラマブルコントローラのプログラム編集装置は、前記PC間インターロック接点指示がPC間インターロックのプログラマブルコントローラ固有名とインターロック条件となるデバイスの記述部を有し、前記元シーケンスプログラムにおける

PC間インターロック接点指示を抽出し、PC間インターロック接点指示のデータより前記共有情報収集テーブルを自動作成するものである。

【0020】つぎの発明によるプログラマブルコントローラのプログラム編集装置は、複数のプログラマブルコントローラが連携するプログラマブルコントローラで使用するシーケンスプログラムを作成するプログラム編集装置において、デバイスが属するプログラマブルコントローラを特定する情報を登録するプログラマブルコントローラ固有名と、インターロック条件となるデバイスの名前を登録する共有情報収集テーブルをメモリに有し、各ステップ毎に各プログラマブルコントローラの固有名を付けられ、トランジション条件としてPC間インターロック接点を記述された一つの順次実行型プログラムのステップに前記共有情報収集テーブルに登録されているデータに基づいて外部接点と共有情報接点を含むトリガープログラムを自動的に付加し、トランジション条件として該当するステップ内のシーケンスプログラムに前記トリガープログラムの共有情報接点をPC間インターロック接点として付加し、前記一つの順次実行型のプログラムを各ステップ毎に分離して各プログラマブルコントローラのシーケンスプログラムに自動的に変換するプログラム作成処理を行うものである。

【0021】つぎの発明によるプログラマブルコントローラのプログラム編集装置は、各ステップ毎に付けられたプログラマブルコントローラの固有名とトランジション条件として一つの順次実行型のプログラムに記述されたPC間インターロック接点とから、前記共有情報収集テーブルを自動作成するものである。

【0022】つぎの発明によるプログラマブルコントローラのプログラム編集装置は、前記共有情報収集テーブルのデータ内容より各プログラマブルコントローラ毎の共有情報接点のデバイス範囲を自動検出し、各プログラマブルコントローラ毎の共有情報接点のデバイス範囲を示す共有情報メモリ定義テーブルを自動作成するものである。

#### 【0023】

【発明の実施の形態】以下に添付の図を参照して、この発明にかかるプログラマブルコントローラのプログラム編集装置の実施の形態を詳細に説明する。

【0024】実施の形態1. 図1はこの発明によるプログラマブルコントローラのプログラム編集装置（周辺装置）のハード構成を示している。プログラム編集装置10は、CPU等を含む処理部11と、メモリ12と、補助記憶装置制御部13と、キー入力制御部14と、画面制御部15とを有し、補助記憶装置制御部13には補助記憶装置16が、キー入力制御部14にはキーボード17が、画面制御部15にはCRT18がそれぞれ接続されている。

【0025】プログラム編集装置10のメモリ12は、

プログラム編集装置10がネットワークに接続されているPCを把握するためのPC固有名管理テーブル20と、後述の共有情報収集テーブル21および共有情報メモリマップ定義テーブル22を格納する。

【0026】PC固有名管理テーブル20は、図2に示されているように、ネットワークに接続されている総PC数のデータ部20aと、各PCに固有に付けられるPC固有名を示すデータ部20bとを割り付けられている。ネットワークに接続されている総てのPCについて、ユーザが任意に付けたPC固有名を入力することによって、プログラム編集装置10でデータが格納される。

【0027】図3はプログラム編集装置10に編集される元シーケンスプログラム例を示している。この元シーケンスプログラムは、ユーザが他のPC（ここではPC1）の共有情報によってインターロックをかけることを指定したPC2のシーケンスプログラムである。

【0028】図3において、120はプログラム編集装置10で利用できるPC間インターロック接点指示を、121は外部接点を、122は内部接点を、123はプログラム編集装置10で利用できるPC間インターロック終了指示をそれぞれ示している。PC間インターロック接点指示120は、インターロックをかけるPCの固有名とそのPCの外部接点の記述部を有する。なお、PC1、PC2は、図23に示されているPCシステムのものと同等のものである。

【0029】共有情報収集テーブル21は、プログラム編集装置10がネットワークに接続されているPC間で他のPCのプログラムのインターロック条件となるデバイスを把握するために、メモリ12に記憶されるものであり、図4に示されているように、デバイスが属するPCを特定する情報を登録するPC固有名列21aと、インターロック条件となるデバイスの名前を登録するデバイス列21bと、PC固有名とデバイスで特定されるPC間インターロック接点情報がプログラムに反映されたかどうかを表すフラグ情報を登録する反映フラグ列21cにより構成されている。

【0030】共有情報収集テーブル21には、図2に示されているようなPC固有名管理テーブル20と、図3に例示されているシーケンスプログラムの内容から、プログラム編集装置10によって自動的にデータが格納される。すなわち、共有情報収集テーブル21は、元シーケンスプログラムにおけるPC間インターロック指示を抽出することで、PC間インターロック接点指示のデータより自動作成される。

【0031】プログラム編集装置10は、共有情報収集テーブル21のデータに基づいて、PC1用のシーケンスプログラムに、図5に示されているように、外部接点(X0)130と共有情報接点(B0)を含むトリガープログラムを自動的に付加する。プログラム編集装置1

0は、PC間インターロック接点指示120からPC間インターロック終了指示123までのすべてのシーケンスプログラムを、PC間インターロック接点指示120で指定されたPCの外部接点によってインターロックがかかるように自動変換する。これは、該当範囲のシーケンスプログラムにトリガープログラムの共有情報接点をPC間インターロック接点として自動的に付加することにより行われる。

【0032】図6は、図3に示したPC2用のシーケンスプログラムをプログラム編集装置10によって自動的に変換したシーケンスプログラムを示しており、PC間インターロック接点命令指示120からPC間インターロック終了命令指示123までの各ステップに共有情報接点(B0)が直列に付加される。

【0033】共有情報メモリマップ定義テーブル22は、プログラム編集装置10が、ネットワークに接続されているPC間で共有する情報のデバイス範囲を把握するために、メモリ12に記憶されるものであり、図7に示されているように、PC固有名22aと、各プログラマブルコントローラ毎の共有情報接点のデバイス範囲を示すデバイス範囲列22bとを有する。共有情報メモリマップ定義テーブル22は、共有情報収集テーブル21のデータ内容より各プログラマブルコントローラ毎の共有情報接点のデバイス範囲を自動検出することで、自動生成される。

【0034】プログラム編集装置10は、共有情報メモリマップ定義テーブル22のデータに基づいて従来例で示した親機PCの通信ユニット60<sub>1</sub>と同等の通信ユニットが保有するデータ共有情報メモリ部62の各PCの公開情報部分62<sub>1</sub>、62<sub>2</sub>・・・のマッピングを図8に示されているように、自動的に行う。換言すれば、データ共有情報メモリマップを自動生成する。

【0035】つぎに、動作について説明する。図23に示されているようなネットワークシステムの構成が存在していたとする。まず、ネットワーク上に存在する全てのPCにユーザが任意の固有名を付ける。ユーザ入力によって、プログラム編集装置10で、ネットワーク上に存在するPCの総数が図2に示されているPC固有名管理テーブル20の総PC数20aに格納され、PC固有名20bにネットワーク上に存在する全てのPCの固有名が登録される。

【0036】2台目のPC2用に図3に示されているような元シーケンスプログラムを作成したとする。この元シーケンスプログラムにおけるPC間インターロック条件の指定方法はユーザが理解しやすくするために定めた方法なので、このままでは実際にPCで使用することができない。

【0037】そこでPCで期待する動作が実行できるように、プログラム編集装置10にも図5、図6に示されているようなシーケンスプログラムを作成する変換処理

を実施する。この変換処理は、ユーザがキーボード17により変換指示を出すことにより開始される。この変換処理は、図9に示されているフローチャートに従って行われる。

【0038】変換処理が始まると、まず、共有情報収集テーブル21の作成が行われる(ステップS150)。共有情報収集テーブル21の作成は図10に示されているフローチャートに従って行われる。共有情報収集テーブル21の作成が終わると、つぎに、共有情報メモリマップ定義テーブル22の作成が行われる(ステップS151)。共有情報メモリマップ定義テーブル22の作成は図11に示されているフローチャートに従って行われる。

【0039】共有情報メモリマップ定義テーブル21の作成が終わると、共有情報収集テーブル21の反映フラグを全てを未反映状態で初期化することが行われ(ステップS152)、この後にインターロック部プログラム代入処理(ステップS153)が行われる。インターロック部プログラム代入処理は図12に示されているフローチャートに従って行われる。なお、ここでは、未反映状態は反映フラグ=“0”とし、この共有情報に関する処理がプログラムに反映済みの状態でのフラグ=“1”と区別することにする。

【0040】共有情報収集テーブル作成を図10に示されているフローチャートを参照して説明する。まず、登録数カウンタN=0とする(ステップS200)。つぎに、PC固有名管理テーブル20に登録されている全てのPC用のプログラムのチェックが出来たかをチェックし(ステップS201)、全てのプログラムのチェックが出来た場合には、終了する。

【0041】全てのプログラムのチェックが終了していない場合には、共有情報収集のチェックしている部分がプログラムの最後であるかを判別する(ステップS202)。プログラムの最後の場合には、つぎのプログラムのチェックを行う。チェックしているプログラムの最後ではなく、途中の場合には、チェック対象がPC間インターロック接点指示120の書式の命令かどうかをチェックする(ステップS203)。PC間インターロック接点指示120ではない場合には、共有情報収集テーブルに取得する情報はないものとして、ステップS202に戻り、プログラムのつぎの部分のチェックを行う。

【0042】PC間インターロック接点120の場合には、PC間インターロック接点120の記述からPC固有名を抽出してそのPC固有名を共有情報収集テーブル21のN行目のPC固有名データフィールドに登録する(ステップS204)。つぎに、PC間インターロック接点指示120から外部接点を抽出して、共有情報収集テーブル21のN行目のデバイスデータフィールドに登録する(ステップS205)。

【0043】つぎに、共有情報収集テーブル21のN行

目の反映フラグを未反映状態で初期化する（ステップS 206）。これにより、共有情報収集テーブル21への一つのPC間インターロック接点120の情報の設定が終了したので、つぎに、登録デバイス数カウンタ $N=N+1$ を行い（ステップS 207）、つぎの情報を収集するために、ステップS 202に戻る。

【0044】すべてのPCのプログラムについてチェックが完了すると、このネットワークに接続されているPCが共有する必要のあるデバイス名の情報が格納された共有情報収集テーブル21が完成する。

【0045】共有情報メモリマップ定義テーブル作成を図11に示されているフローチャートを参照して説明する。まず、ステップ250でPC別のデバイス使用数カウンタ $N=0$ とする（ステップS 250）。つぎに、PC固有名管理テーブル20に登録されている全てのPCのためのプログラムについてチェックが行われたかをチェックする（ステップS 251）。全てのPCのプログラムがチェック済みの場合には、終了する。

【0046】全てのPCのプログラムのチェックが行われていない場合には、まだチェックしていないPCのプログラムについて、共有情報収集テーブル21を最初から参照してチェックするために、共有情報収集テーブル21の参照する参照登録情報番号カウンタ $J=0$ とする（ステップS 252）。

【0047】つぎに、参照登録情報番号カウンタ $J$ が共有情報収集テーブル21の最後かをチェックする（ステップS 253）。参照登録情報番号カウンタ $J$ が共有情報収集テーブルの最後の場合には、この固有名のPCに関する共有情報収集テーブル21のチェックが終了したことになるので、共有情報メモリマップ定義テーブル22のPC固有名データフィールドにPC固有名を、デバイス範囲データフィールドにデバイス使用数として $N$ を登録し、つぎのPC固有名のチェックを行うため、ステップS 250に戻る。

【0048】参照登録情報番号カウンタ $J$ が共有情報収集テーブルの最後ではない場合には、共有情報収集テーブル21の参照登録情報番号カウンタ $J$ の行の反映フラグが“0”で未反映かどうかをチェックする（ステップS 254）。共有情報収集テーブル21の参照登録情報番号カウンタ $J$ の行の反映フラグが“1”の場合には、既に該当行の情報は反映されているとして、参照登録情報番号カウンタ $J=J+1$ とし（ステップS 263）、ステップS 253に戻る。

【0049】これに対し、共有情報収集テーブル21の参照登録情報番号カウンタ $J$ の行の反映フラグが“0”の場合には、まだこの共有情報のデバイスに関する反映が行われていないので、共有情報収集テーブル21の参照登録情報番号カウンタ $J$ の位置のPC固有名が現在チェック中のPC固有名かをチェックする（ステップS 255）。共有情報収集テーブル21の参照登録情報番号

カウンタ $J$ の行のPC固有名が現在チェック中のPC固有名ではない場合にはこの共有情報は無視して参照登録情報番号カウンタ $J=J+1$ とし（ステップS 263）、ステップS 253に戻る。

【0050】共有情報収集テーブル21の参照登録情報番号カウンタ $J$ の行のPC固有名が現在チェック中のPC固有名の場合には、このPCに関する共用が必要な新規デバイスに関する情報があつたとし、デバイス使用数 $N=N+1$ とする（ステップS 256）。

10 【0051】つぎに、共有情報収集テーブル21の反映フラグに既に情報を反映したことを示すために“1”を代入する（ステップS 257）。そして、このデバイスと同じデバイスに関する情報を共有デバイス使用数 $N$ として二重にカウントしないために、共有情報収集テーブル21の参照登録情報番号カウンタ $J$ 以降で、同じデバイスに関する情報があるかどうか検索すべく、同一デバイス検索用登録情報番号カウンタ $I$ を設定して $I=J+1$ とする（ステップS 258）。

20 【0052】つぎに、同一デバイス検索用登録情報番号カウンタ $I$ が共有情報収集テーブル21の最後かをチェックする（ステップS 259）。同一デバイス検索用登録情報番号カウンタ $I$ が共有情報収集テーブル21の最後の場合には、同デバイスの検索が終了したことになるので、参照登録情報番号カウンタ $J=J+1$ とし（ステップS 263）、ステップS 253に戻る。

30 【0053】これに対し、同デバイス検索用登録情報番号カウンタ $I$ が共有情報収集テーブル21の最後ではない場合には、参照登録情報番号カウンタ $J$ と同一デバイス検索用登録情報番号カウンタ $I$ の位置の共有情報収集テーブル21のデバイスが同じかをチェックする（ステップS 260）。デバイスが異なる場合は、同一デバイス検索用登録情報番号カウンタ $I=I+1$ とし（ステップS 264）、ステップS 259に戻る。

40 【0054】共有情報収集テーブル21のデバイスが参照登録情報番号カウンタ $J$ と同一デバイス検索用登録情報番号カウンタ $I$ の情報で同じ場合には、同デバイス検索用登録情報番号カウンタ $I$ の情報については、すでに参照登録情報番号カウンタ $J$ で反映は完了したとみなし、同一デバイス検索用登録情報番号カウンタ $I$ の行の共有情報収集テーブルの反映フラグに、既に反映済みとして“1”を設定する（ステップS 261）。

【0055】つぎに、同一デバイス検索用登録情報番号カウンタ $I=I+1$ とし（ステップS 264）、ステップS 259に戻る。すべてのPC固有名について共有情報収集テーブル21のチェックが完了すると、このネットワークに接続されているPCについての共有するデバイスの必要範囲情報が格納された共有情報メモリマップ定義テーブル22が自動的に作成される。

【0056】インターロック部プログラム代入処理を図12に示されているフローチャートを参照して説明す



る。まず、共有デバイス番号カウンタ $N=0$ とする（ステップS300）。つぎに、全てのPC用のプログラムのチェックが出来たかをチェックする（ステップS301）。全てのプログラムのチェックが終了した場合には、処理を終了する。全てのプログラムのチェックが終了していない場合には、プログラムのチェック位置カウンタ $J=0$ とする（ステップS302）。

【0057】つぎに、チェックしている部分がプログラムの最後かをチェックする（ステップS303）。チェックしている部分がプログラムの最後の場合には、トリガープログラム代入を行う（ステップS311）。トリガープログラム代入処理は図13に示されているフローチャートに従って行われる。トリガープログラム代入処理により共有デバイス番号カウンタ $N$ が示す共有デバイスに関する処理が終了すると、共有デバイス番号カウンタ $N=N+1$ を行い（ステップS310）、ステップS301に戻る。

【0058】チェックしている部分がプログラムの最後ではない場合には、チェック部分がPC間インターロック接点指示120の書式かどうかをチェックする（ステップS304）。PC間インターロック接点指示120ではない場合には、プログラムチェック位置カウンタ $J=J+1$ を行い（ステップS312）、ステップS303に戻り、プログラムのつぎの部分のチェックを行う。

【0059】PC間インターロック接点指示120の場合には、つぎに、PC間インターロック終了指示123かどうかのチェックを行う（ステップS305）。PC間インターロック終了指示123の場合には、プログラムチェック位置カウンタ $J=J+1$ を行い（ステップS312）、ステップS303に戻り、プログラムのつぎの部分のチェックを行う。

【0060】PC間インターロック終了指示123でない場合には、プログラムチェック位置カウンタ $J=J+1$ を行い（ステップS306）、PC間インターロック接点指示120からPC間インターロック終了指示123までの間のプログラムにはすべてPC間インターロック接点で指定された条件でインターロックをかける必要がある。ラダー図の左側の線に付いている外部接点121の前に共有情報接点140を追加しなければならない。そこで、つぎに、ラダー図の左側の線に付いている外部接点121かどうかをチェックする（ステップS307）。左側の線に付いている外部接点121ではない場合には、ステップS305に戻る。

【0061】これに対し、左側の線に付いている外部接点121の場合には、現在の部分に共有情報接点140を挿入する（ステップS308）。この挿入により、プログラム先頭からの位置カウンタ $J$ がずれるので、プログラムチェック位置カウンタ $J=J+1$ を行い（ステップS309）、ステップS305に戻る。すべてのPCのプログラムについてチェックが完了すると、このネッ

トワークに接続されているPCのプログラムにPC間のインターロックに関する命令の挿入が完了したことになる。

【0062】インターロックのトリガープログラム代入処理を図13に示されているフローチャートを参照して説明する。まず、共有情報収集テーブル21の参照登録番号カウンタ $J=0$ とする（ステップS350）。つぎに、参照登録番号カウンタ $J$ が共有情報収集テーブル21の最後かをチェックする（ステップS351）。共有情報収集テーブル21の最後の場合には処理を終了する。

【0063】参照登録番号カウンタ $J$ が共有情報収集テーブル21の最後ではない場合には、共有情報収集テーブル21の参照登録番号カウンタ $J$ の行の反映フラグが未反映“0”かをチェックする（ステップS353）。反映フラグが未反映“0”でない場合には、参照登録番号カウンタ $J=J+1$ として（ステップS361）、ステップS351に戻る。

【0064】共有情報収集テーブル21の $J$ の行の反映フラグ131が“0”で未反映の場合には、共有情報収集テーブル21の $J$ の行のPC固有名が現在チェック中のPC固有名かどうかをチェックする（ステップS354）。現在チェック中のPC固有名ではない場合には、参照登録番号カウンタ $J=J+1$ として（ステップS361）、ステップS351に戻る。

【0065】現在チェック中のPC固有名の場合には、プログラムの最後に外部接点130オンに対して共有情報接点131をオンする図5に示されているようなトリガープログラムを追加する（ステップS355）。これにより、共有情報収集テーブル21の $J$ の行の共有デバイス情報についてプログラムへの反映処理が終了したので、参照登録番号カウンタ $J$ の位置の共有情報収集テーブル21の反映フラグに“1”を代入する（ステップS356）。

【0066】そして、このデバイスと同じデバイスに関するトリガープログラムを二重に挿入しないために、共有情報収集テーブル21の参照登録情報番号カウンタ $J$ 以降で、同一デバイス検索用登録情報番号カウンタ $I$ を設定し、同じデバイスに関する情報があるかどうか検索すべく、 $I=J+1$ とする（ステップS357）。

【0067】つぎに、同一デバイス検索用登録情報番号カウンタ $I$ が共有情報収集テーブル21の最後かをチェックする（ステップS358）。同一デバイス検索用登録情報番号カウンタ $I$ が共有情報収集テーブル21の最後の場合には、同デバイスに関する検索が終了したものとして、参照登録番号カウンタ $J=J+1$ として（ステップS361）、ステップS351に戻る。

【0068】同一デバイス検索用登録情報番号カウンタ $I$ が共有情報収集テーブル21の最後ではない場合には、参照登録番号カウンタ $J$ と同一デバイス検索用登録

情報番号カウンタ I の位置の共有情報収集テーブル 2 1 のデバイスと PC 固有名が同じかをチェックする (ステップ S 3 5 9)。異なる場合には、同一デバイス検索用登録情報番号カウンタ  $I = I + 1$  として (ステップ S 3 6 2)、ステップ S 3 5 8 に戻る。同じ場合には、同一デバイス検索用登録情報番号カウンタ I の行の共有情報収集テーブル 2 1 の反映フラグに “1” を代入し (ステップ S 3 6 0)、この情報に関する反映処理は完了したことにして同一デバイス検索用登録情報番号カウンタ  $I = I + 1$  とし (ステップ S 3 6 2)、ステップ S 3 5 8 に戻る。

【0069】すべての共有情報についての反映が完了すると、インターロックに必要な共有デバイスのトリガープログラムがすべてそれぞれの PC のプログラム末に挿入され、ユーザがインターロック接点で設定した内部接点のデバイスがオンすることによって、自動的に割り付けた対応する共有デバイスがオンすることになる。

【0070】対応する共有デバイスに関するインターロックのための共有情報接点 1 4 0 は、インターロック部プログラム代入処理ルーチンによって自動挿入されているため、プログラムを実行したとき、ユーザは共有デバイスを意識しなくても、自分が指定した PC 間インターロック接点のデバイスのオン/オフによって、PC 間インターロック終了指示までの間のプログラムにインターロックがかかる。

【0071】自動的に割り付けた共有デバイスの範囲については、共有情報メモリマップ定義テーブル作成ルーチンにより共有情報メモリマップ定義テーブル 2 2 に格納されているので、必要に応じてこれを参照することができる。

【0072】実施の形態 2。図 1 4 はこの発明によるプログラマブルコントローラのプログラム編集装置 (周辺装置) のハード構成を示している。プログラム編集装置 3 0 は、CPU 等を含む処理部 3 1 と、メモリ 3 2 と、補助記憶装置制御部 3 3 と、キー入力制御部 3 4 と、画面制御部 3 5 とを有し、補助記憶装置制御部 3 3 には補助記憶装置 3 6 が、キー入力制御部 3 4 にはキーボード 3 7 が、画面制御部 3 5 には CRT 3 8 がそれぞれ接続されている。プログラム編集装置 3 0 のメモリ 3 2 は、後述の共有情報収集テーブル 4 1 および共有情報メモリマップ定義テーブル 4 2 を格納する。

【0073】図 1 5 はプログラム編集装置 3 0 により編集される順次実行型プログラムの 1 例を示している。この順次実行型プログラムは、複数台の PC のために、従来ならば、各 PC 毎に分けて書かねばならないシーケンスプログラムを一本につなげて書いたものである。図 1 5 に例示されている順次実行型プログラムは、1 台目の PC のプログラムにあたるイニシャルステップ (処理 1) と、2 台目の PC のプログラムにあたるステップ (処理 2) と、3 台目の PC のプログラムにあたるステ

ップ (処理 3) を一本のプログラムとして記述されている。

【0074】各ステップには、個別のラベル S 1、S 2、S 3 が付帯し、その各ラベル S 1、S 2、S 3 にシーケンスを実行する PC 固有名が付属している。各ステップにはステップ内シーケンス A、B、C がある。

【0075】処理 1 のステップ内シーケンス A のつぎには、処理 1 から処理 2 に移る条件判定となるトランジション T 2 と、PC 1 と PC 2 の間のインターロックのトリガとなるトランジション内 PC 間インターロック接点 M 0 がある。

【0076】処理 2 のステップ内シーケンス B のつぎには、処理 2 から処理 3 に移る条件判定となるトランジション T 3 と、PC 2 と PC 3 の間のインターロックのトリガとなるトランジション内 PC 間インターロック接点 M 1 0 がある。

【0077】共有情報収集テーブル 4 1 は、プログラム編集装置 3 0 がネットワークに接続されている PC 間で他の PC のプログラムのインターロック条件となるデバイスを把握するために、メモリ 3 2 に記憶されるものであり、図 1 6 に示されているように、デバイスが属する PC を特定する情報を登録する PC 固有名列 4 1 a と、インターロック条件となるデバイス (接点) の名前を登録する PC 間インターロック接点列デバイス列 4 1 b により構成されている。

【0078】共有情報収集テーブル 4 1 は、各ステップ毎に付けられた PC 固有名とトランジション条件として一つの順次実行型のプログラムに記述された PC 間インターロック接点とから自動作成される。プログラム編集装置 3 0 は、共有情報収集テーブル 4 1 を参照して行うデータに各 PC シーケンス作成処理により、上述のように記述された順次実行型プログラムを図 1 7 (a) ~ (c) に示されているような各 PC 用のシーケンスプログラムに分割変換する。

【0079】図 1 7 (a) は PC 1 のシーケンスプログラムであり、ステップ内シーケンス A に常時閉接点 (ダミー接点) M 9 0 3 6 を自動付加され、最後にトランジション T 2 に対応する PC 1 インターロック条件シーケンス (トリガープログラム) 2 0 0 を自動付加される。PC 1 インターロック条件シーケンス 2 0 0 は、内部接点 M 0 と共有情報接点 B 0 とを有する。

【0080】図 1 7 (b) は PC 2 のシーケンスプログラムであり、ステップ内シーケンス B に共有情報接点 B 0 を自動付加され、最後にトランジション T 3 に対応する PC 2 インターロック条件シーケンス (トリガープログラム) 2 0 1 を自動付加される。PC 2 インターロック条件シーケンス 2 0 1 は、内部接点 M 1 0 と共有情報接点 B 1 0 とを有する。

【0081】図 1 7 (c) は PC 3 のシーケンスプログラムであり、ステップ内シーケンス C に共有情報接点 B

10を自動付加される。

【0082】なお、プログラム編集装置30は、各シーケンスプログラムを編集するためにメモリ32に領域を確保し、各シーケンスの格納領域を設ける。

【0083】図18は親機PCの通信ユニット60<sub>1</sub>のメモリ61（図23参照）に展開されるデータ共有情報メモリマップを示している。共有情報はネットワークで接続されたPC間で共有される情報（共有情報接点）であり、このメモリマップは、1台目のPCが公開する情報を格納するためのPC1占有領域（PC1公開情報部分）、2台目のPCが公開する情報を格納するためのPC2占有領域（PC2公開情報部分）など、各PCの公開情報のための占有領域から成り立っている。

【0084】共有情報メモリマップ定義テーブル42は、プログラム編集装置10が、ネットワークに接続されているPC間で共有する情報のデバイス範囲を把握するために、メモリ32に記憶されるものであり、図19に示されているように、PC固有名列42aと、デバイス範囲列42bとを有する。共有情報メモリマップ定義テーブル42は、共有情報収集テーブル41のデータ内容より各プログラマブルコントローラ毎の共有情報接点のデバイス範囲を自動検出することで、自動生成される。

【0085】プログラム編集装置30は、共有情報メモリマップ定義テーブル42のデータに基づいて従来例で示した親機PCの通信ユニット60<sub>1</sub>と同等の通信ユニットが保有するデータ共有情報メモリ部62の各PCの公開情報部分62<sub>1</sub>、62<sub>2</sub>・・・のマッピングを図18に示されているように、自動的に行う。換言すれば、データ共有情報メモリマップを自動生成する。

【0086】つぎに、動作について説明する。実施の形態2でも、図23に示されているようなネットワークシステムの構成が存在していたとする。まず、ユーザ入力によって、プログラム編集装置30で、ネットワーク上に存在する全てのPCに任意の固有名を設定する。つぎに、ユーザは、プログラム編集装置30により図15に示されているような複数PC用の順次実行型プログラムを作成する。

【0087】つぎに、プログラム編集装置30において共有情報収集テーブル作成処理を実行する。共有情報収集テーブル作成処理は図20に示されているフローチャートに従って行われる。まず、順次実行型プログラムの先頭のイニシャルステップへプログラム内容の判定位置を移動する（ステップS400）。つぎに、判定位置に該当するステップがエンドステップであるかを調べる（ステップS401）。最初にこのステップへ来る場合には必ずエンドステップではない。

【0088】エンドステップでない場合には、該当するステップのラベルに付加しているPC固有名（例えば、“PC1”）を共有情報収集テーブル41のPC固有名

列41aの空行に格納する（ステップS402）。

【0089】つぎに、トランジションへ移り（ステップS403）、トランジションの中にあるトランジション内PC間インターロック条件接点にあるデバイス（例えば、“M0”）を共有情報収集テーブル41のPC間インターロック接点列41bの該当行（“PC1”の格納行と同一の行）に格納する（ステップS404）。

【0090】つぎに、複数PC用順次実行型プログラムのつぎのステップへ移動し、ステップS401に戻る。エンドステップになれば、共有情報収集テーブル41のデータ格納行をPC間インターロック接点列のデバイスの番号昇順に入れ替えを行う（ステップS405）。

【0091】上述の処理により図16に示されているような共有情報収集テーブル41が自動作成される。共有情報収集テーブル41の自動作成が完了すれば、共有情報メモリマップ定義テーブル作成処理へ進む（ステップS406）。

【0092】共有情報メモリマップ定義テーブル作成処理は図21に示されているフローチャートに従って行われる。まず、共有情報収集テーブル41の一番上の行に判定位置を移動する（ステップS450）。つぎに、共有情報収集テーブル41の行にデータが設定されているかを調べる（ステップS451）。

【0093】データが設定されていれば、共有情報収集テーブル41のPC固有名列の現在行の固有名が前の行の固有名と違うかを調べる（ステップS452）。固有名が異なる場合には、共有情報メモリマップ定義テーブル42のまだPC固有名が設定されていない空行のPC固有名データフィールドにPC固有名を設定し、その行のデバイス範囲データフィールドの先頭に共有情報接点番号として共有情報収集テーブル41におけるPC間インターロック接点番号と同じ番号を設定する（ステップS453）。この後に、共有情報収集テーブル41のつぎの行へ移り、ステップS451に戻る。

【0094】ステップS452の判定において、共有情報収集テーブル41のPC固有名列41aの現在行の固有名が前の行の固有名と同じであれば、現在行のデバイス範囲データフィールドの最終に共有情報接点番号として共有情報収集テーブル41におけるPC間インターロック接点番号と同じ番号を設定する（ステップS454）。この後に、共有情報収集テーブル41のつぎの行へ移り、ステップS451に戻る。

【0095】すべてのPC固有名について共有情報収集テーブル41のチェックが完了すると、実施の形態1における場合と同様に、このネットワークに接続されているPCについての共有するデバイスの必要範囲情報が格納された共有情報メモリマップ定義テーブル42が自動的に作成される。

【0096】各PCシーケンス作成処理は図22に示されているフローチャートに従って行われる。まず、実行

位置を複数PC用順次実行型プログラムの先頭ステップであるイニシャルステップへ移動する（ステップS500）。つぎに、そのステップがエンドステップであるかを調べる（ステップS501）。

【0097】エンドステップでない場合には、プログラムのステップのラベルに付属しているPC固有名を見る（ステップS502）。そのPC固有名がメモリ32の割り付けられている各PCシーケンス格納領域のPC固有名と一致しない名前かを調べる（ステップS503）。まだPCシーケンス格納領域がない新規のPC固有名であれば、新しい各PCシーケンス格納領域をメモリ32に作成する（ステップS504）。

【0098】つぎに、複数PC用順次実行型プログラムのステップの中に存在するシーケンスを該当するPCシーケンス領域にコピーする（ステップS505）。つぎに、プログラムのつぎのステップのラベルに付属しているPC固有名を見て（ステップS506）、前のステップと同じPC固有名かを調べる（ステップS507）。同じであれば、ステップS505に戻り、PCシーケンス領域へのコピーを行う。

【0099】これに対し、PC固有名が異なる場合には、ステップの前にあるトランジションの中のトランジション内PC間インターロック条件接点（M0、M10等）を各PCシーケンス格納領域の入力側接点に置き、トランジション内PC間インターロック条件接点のデバイスと同じ番号の共有情報接点（B0、B10等）を出力側へ置いたインターロック条件シーケンスを各PCシーケンス格納領域のシーケンスプログラム末へ付加する（ステップS508）。

【0100】つぎに、プログラムのステップの前にあるトランジションのトランジション内PC間インターロック条件接点（B0、B10等）をインターロック条件シーケンスの位置へ付加する（ステップS509）。これが完了すれば、ステップS501に戻り、エンドステップになるまで、ステップS502～ステップS509が繰り返し実行されることで、図17（a）～（c）に示されているような各PC毎のシーケンスプログラムが自動変換作成される。

#### 【0101】

【発明の効果】以上の説明から理解される如く、この発明によるプログラマブルコントローラのプログラム編集装置によれば、デバイスが属するプログラマブルコントローラを特定する情報を登録するプログラマブルコントローラ固有名と、インターロック条件となるデバイスの名前を登録した共有情報収集テーブルのデータに基づいて、該当するプログラマブルコントローラのシーケンスプログラムに外部接点と共有情報接点を含むトリガープログラムを自動的に付加し、PC間インターロック対象のプログラマブルコントローラのシーケンスプログラムにトリガープログラムの共有情報接点をPC間インター

ロック接点として自動的に付加するから、複数のプログラマブルコントローラが連携するプログラマブルコントローラで使用するシーケンスプログラムの作成効率が向上し、しかも設定ミスのない正確なシーケンスプログラムを確実に作成することができる。

【0102】つぎの発明によるプログラマブルコントローラのプログラム編集装置によれば、PC間インターロック対象のプログラマブルコントローラの元シーケンスプログラムに設定されたPC間インターロック接点指示とPC間インターロック終了指示の間のすべてのプログラムにPC間インターロック接点を自動的に付加するプログラム作成処理を行うから、元シーケンスプログラムにおけるPC間インターロック接点指示とPC間インターロック終了指示の設定により、シーケンスプログラムの所定範囲にPC間インターロック接点を自動的に付加でき、複数のプログラマブルコントローラが連携するプログラマブルコントローラで使用するシーケンスプログラムの作成効率が向上し、しかも設定ミスのない正確なシーケンスプログラムを確実に作成することができる。

【0103】つぎの発明によるプログラマブルコントローラのプログラム編集装置によれば、PC間インターロック接点指示にPC間インターロックのプログラマブルコントローラ固有名とインターロック条件となるデバイスの記述部があつてPC間インターロック接点指示のデータより共有情報収集テーブルを自動作成するから、共有情報収集テーブルの作成の手間がかからず、また共有情報収集テーブルが正確に作成され、設定ミスのない正確なシーケンスプログラムを確実に作成することができる。

【0104】つぎの発明によるプログラマブルコントローラのプログラム編集装置によれば、複数のプログラマブルコントローラが連携するプログラマブルコントローラで使用するシーケンスプログラムを作成するプログラム編集装置において、トランジション条件としてPC間インターロック接点を記述された一つの順次実行型プログラムのステップに共有情報収集テーブルに登録されているデータに基づいて外部接点と共有情報接点を含むトリガープログラムを自動的に付加し、トランジション条件として該当するステップ内のシーケンスプログラムにトリガープログラムの共有情報接点をPC間インターロック接点として付加し、一つの順次実行型のプログラムを各ステップ毎に分離して各プログラマブルコントローラのシーケンスプログラムに自動的に変換するから、複数のプログラマブルコントローラが連携するプログラマブルコントローラで使用するシーケンスプログラムの作成効率が向上し、しかも設定ミスのない正確なシーケンスプログラムを確実に作成することができる。

【0105】つぎの発明によるプログラマブルコントローラのプログラム編集装置によれば、各ステップ毎に付けられたプログラマブルコントローラの固有名とトラン

ジション条件として一つの順次実行型のプログラムに記述されたPC間インターロック接点とから、共有情報収集テーブルを自動作成するから、共有情報収集テーブルの作成の手間がかからず、また共有情報収集テーブルが正確に作成され、設定ミスのない正確なシーケンスプログラムを確実に作成することができる。

【0106】つぎの発明によるプログラマブルコントローラのプログラム編集装置によれば、共有情報収集テーブルのデータ内容より各プログラマブルコントローラ毎の共有情報接点のデバイス範囲を自動検出し、各プログラマブルコントローラ毎の共有情報接点のデバイス範囲を示す共有情報メモリ定義テーブルを自動作成するから、ネットワーク上に接続された個々のPCの共通情報として範囲指定する必要のあった共通情報メモリマップを自動的に生成することができ、プログラム作成の効率が向上する。

#### 【図面の簡単な説明】

【図1】 この発明によるプログラマブルコントローラのプログラム編集装置のハード構成を示すブロック線図である。

【図2】 この発明によるプログラマブルコントローラのプログラム編集装置で使用されるPC固有名管理テーブルの構成図である。

【図3】 この発明によるプログラマブルコントローラのプログラム編集装置で作成される元シーケンスプログラムの1例を示すラダー回路図である。

【図4】 この発明によるプログラマブルコントローラのプログラム編集装置で使用される共有情報収集テーブル構成図である。

【図5】 トリガープログラムのラダー回路図である。

【図6】 この発明によるプログラマブルコントローラのプログラム編集装置による変換後のシーケンスプログラムのラダー回路図である。

【図7】 この発明によるプログラマブルコントローラのプログラム編集装置で作成される共有情報メモリマップ定義テーブルの構成図である。

【図8】 共有情報メモリマップの構成図である。

【図9】 この発明によるプログラマブルコントローラのプログラム編集装置におけるゼネラル処理フローを示すフローチャートである。

【図10】 この発明によるプログラマブルコントローラのプログラム編集装置における共有情報収集テーブル作成処理フローを示すフローチャートである。

【図11】 この発明によるプログラマブルコントローラのプログラム編集装置における共有情報メモリマップ定義テーブル作成処理フローを示すフローチャートである。

【図12】 この発明によるプログラマブルコントローラのプログラム編集装置におけるインターロック部プログラム代入処理フローを示すフローチャートである。

【図13】 この発明によるプログラマブルコントローラのプログラム編集装置におけるトリガープログラム代入処理フローを示すフローチャートである。

【図14】 この発明によるプログラマブルコントローラのプログラム編集装置のハード構成を示すブロック線図である。

【図15】 この発明によるプログラマブルコントローラのプログラム編集装置で作成される複数PC用順次実行型プログラムの1例を示すラダー回路図である。

【図16】 この発明によるプログラマブルコントローラのプログラム編集装置で使用される共有情報収集テーブル構成図である。

【図17】 (a)～(c)はこの発明によるプログラマブルコントローラのプログラム編集装置による各PCシーケンスプログラム作成処理により分割作成された各PCのシーケンスプログラムを示すラダー回路図である。

【図18】 共有情報メモリマップの構成図である。

【図19】 この発明によるプログラマブルコントローラのプログラム編集装置で作成される共有情報メモリマップ定義テーブルの構成図である。

【図20】 この発明によるプログラマブルコントローラのプログラム編集装置における共有情報収集テーブル作成処理フローを示すフローチャートである。

【図21】 この発明によるプログラマブルコントローラのプログラム編集装置における共有情報メモリマップ定義テーブル作成処理フローを示すフローチャートである。

【図22】 この発明によるプログラマブルコントローラのプログラム編集装置における各PCシーケンスプログラム作成処理フローを示すフローチャートである。

【図23】 互いに連携する複数のPCによって機械設備のシーケンス制御を行うPCリンクシステムの一般例を示すブロック線図である。

【図24】 PC50<sub>1</sub>のシーケンスプログラム(トリガープログラム)を示すラダー回路図である。

【図25】 PC50<sub>2</sub>のインターロック部分のシーケンスプログラムを示すラダー回路図である。

#### 【符号の説明】

10 プログラム編集装置、11 処理部、12 メモリ、13 補助記憶装置制御部、14 キー入力制御部、15 画面制御部、16 補助記憶装置、17 キーボード、18 CRT、20 PC固有名管理テーブル、21 共有情報収集テーブル、22 共有情報メモリマップ定義テーブル、30 プログラム編集装置、31 処理部、32 メモリ、33 補助記憶装置制御部、34 キー入力制御部、35 画面制御部、36 補助記憶装置、37 キーボード、38 CRT、41 共有情報収集テーブル、42 共有情報メモリマップ定義テーブル、50<sub>1</sub>～50<sub>n</sub> PC本体、51<sub>1</sub>～51<sub>n</sub>

21

22

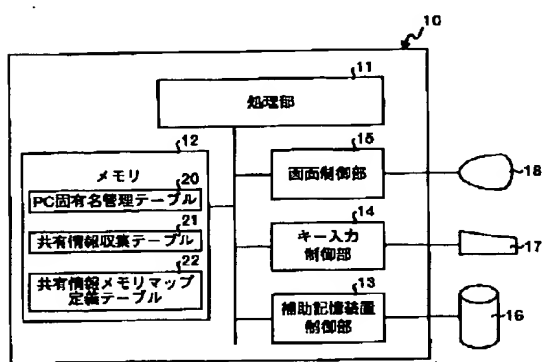
。メモリ、52<sub>1</sub>～52<sub>n</sub>。PC内情報メモリ部、53<sub>1</sub>～53<sub>n</sub>。公開情報部分、60<sub>1</sub>～60<sub>n</sub>。通信ユニット、61メモリ、62データ共有情報メモリ部、120\*

\* PC間インターロック接点指示、123 PC間インターロック終了指示、200, 201 インターロック条件シーケンス（トリガープログラム）。

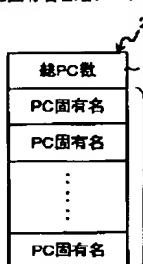
【図 1】

【図 2】

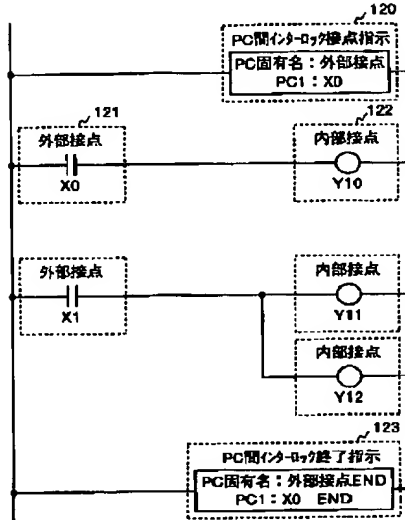
【図 3】



PC固有有名管理テーブル



PC2



【図 4】

共有情報収集テーブル

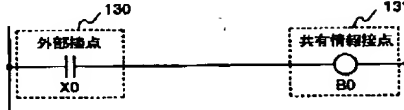
PC固有有名	21a	21b	21c
PC固有名列	デバイス列	反映フラグ列	
行1	PC1	X0	0
行2			
行N			

【図 5】

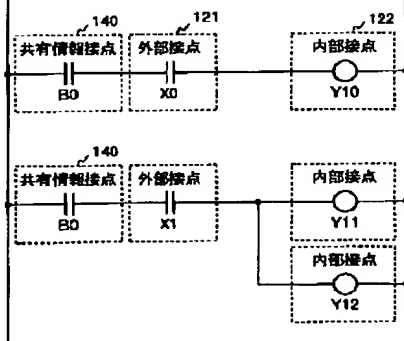
【図 6】

【図 7】

PC1



PC2

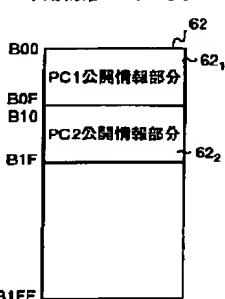


共有情報メモリマップ定義テーブル

行	22a	22b
PC固有名列	デバイス範囲列	
PC1	B0～B0F	
PC2	B10～B1F	
PC名	先頭～最終	

【図 8】

共有情報メモリマップ



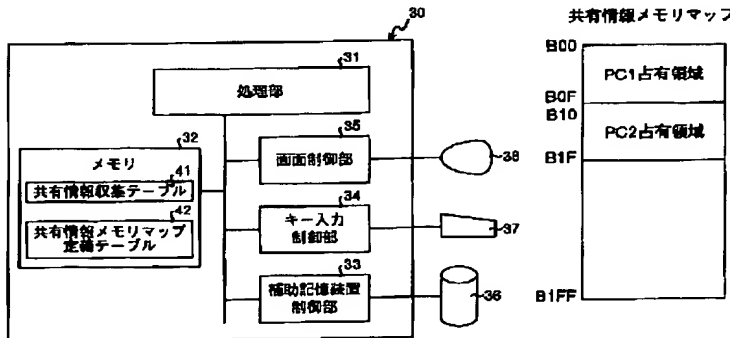
【図 10】

共有情報収集テーブル

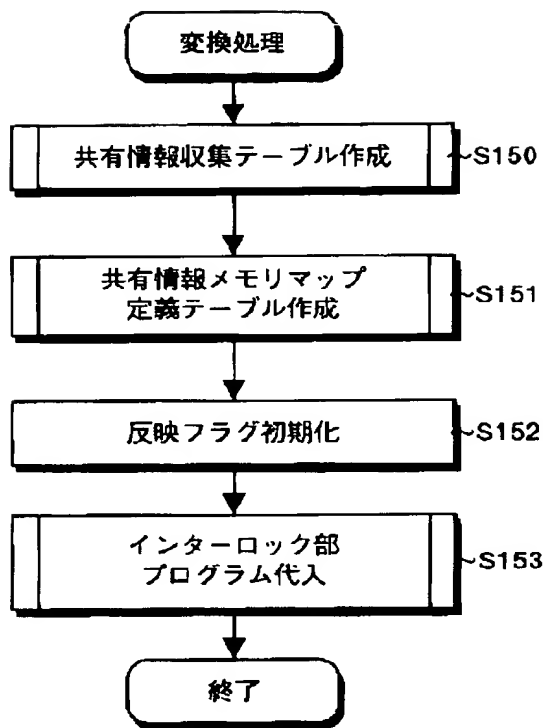
41a	41b
PC固有名列	PC間インターロック接点列
PC1	M0
PC2	M10

【図 14】

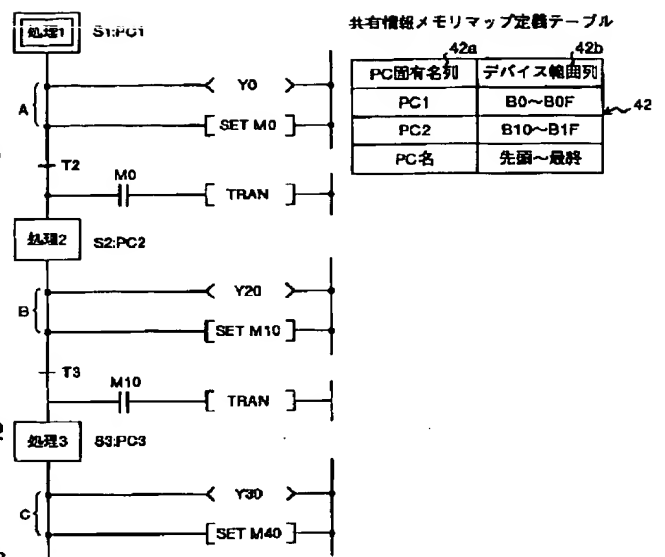
【図 18】



【図 9】



【図 15】

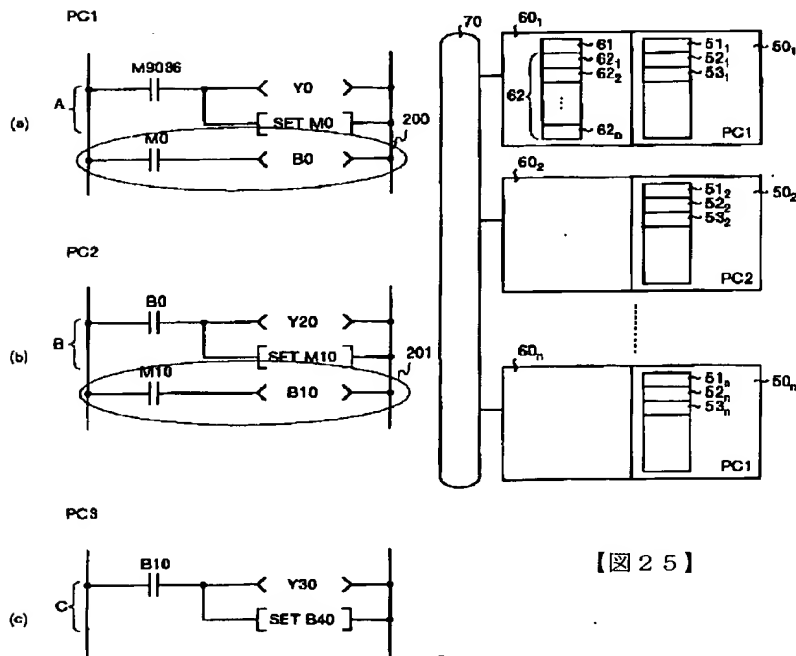


【図 24】

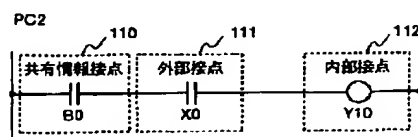


【図 17】

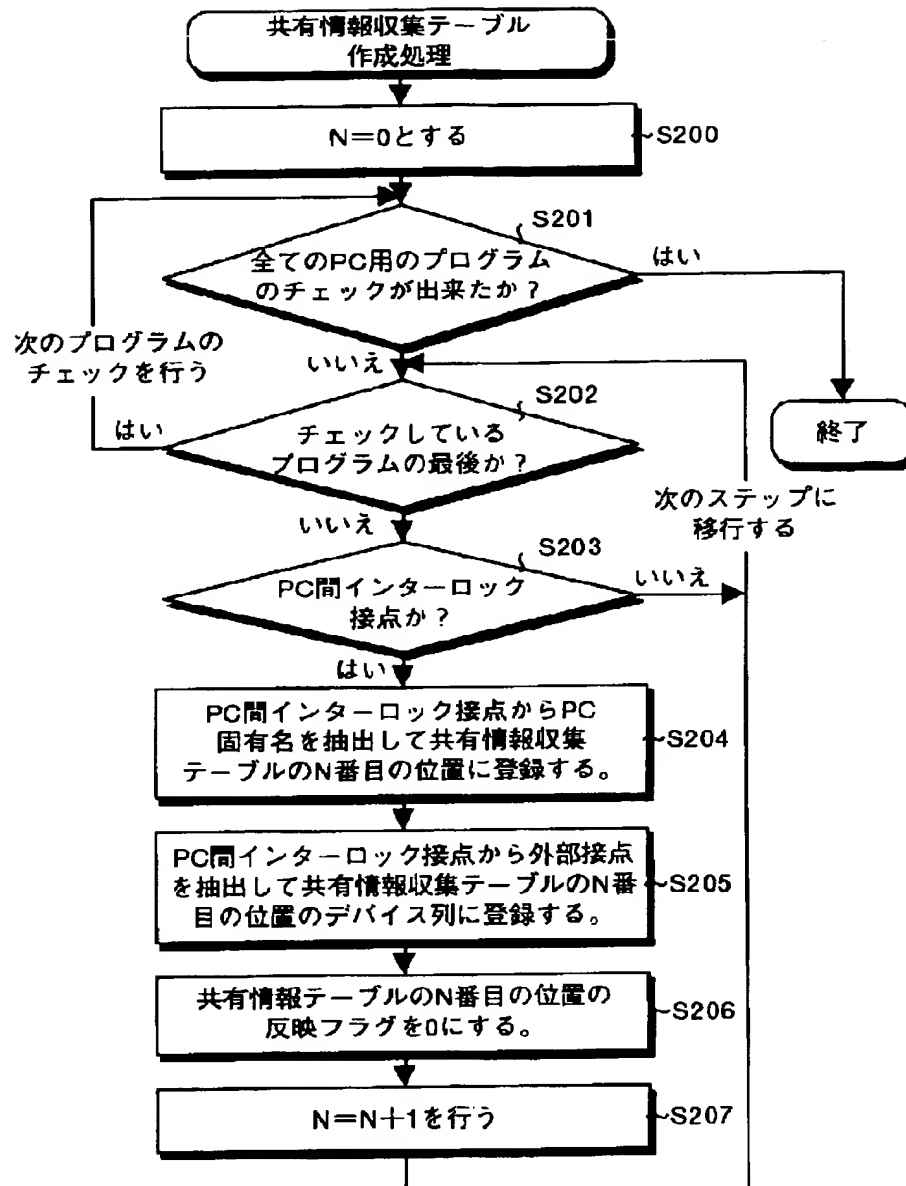
【図 23】



【図 25】

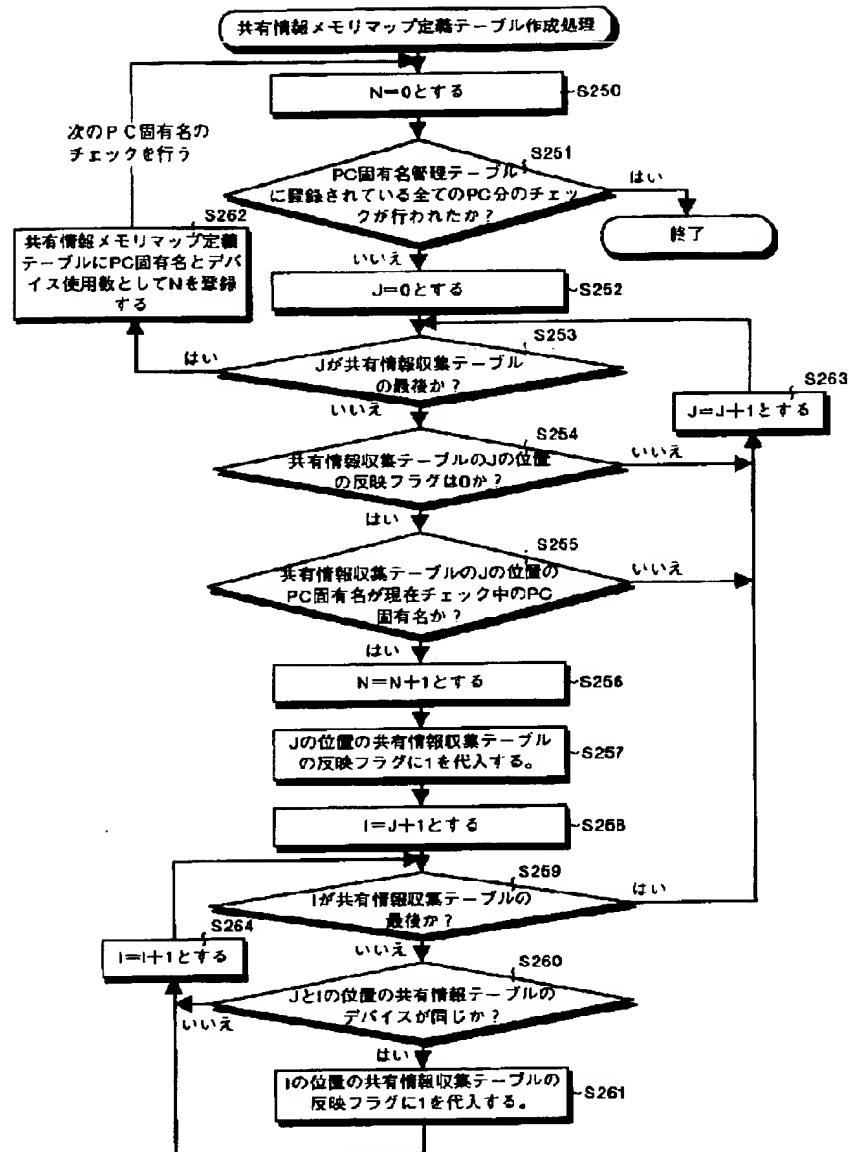


【図10】

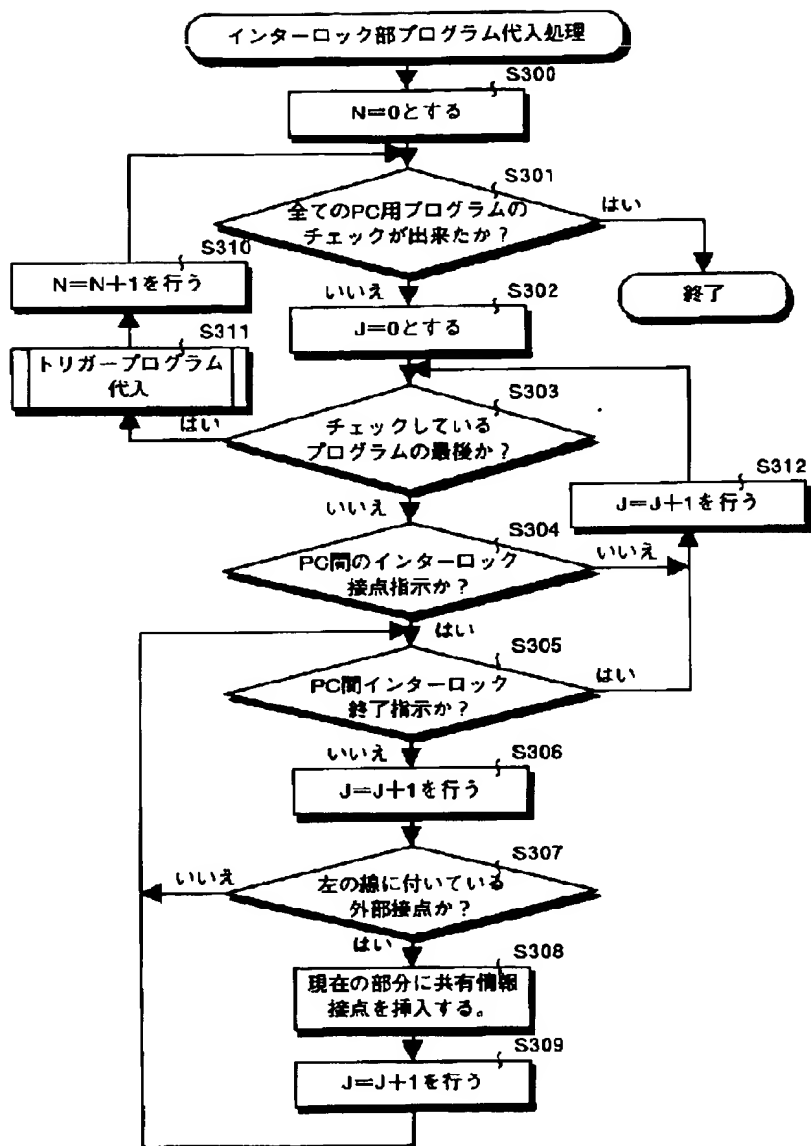




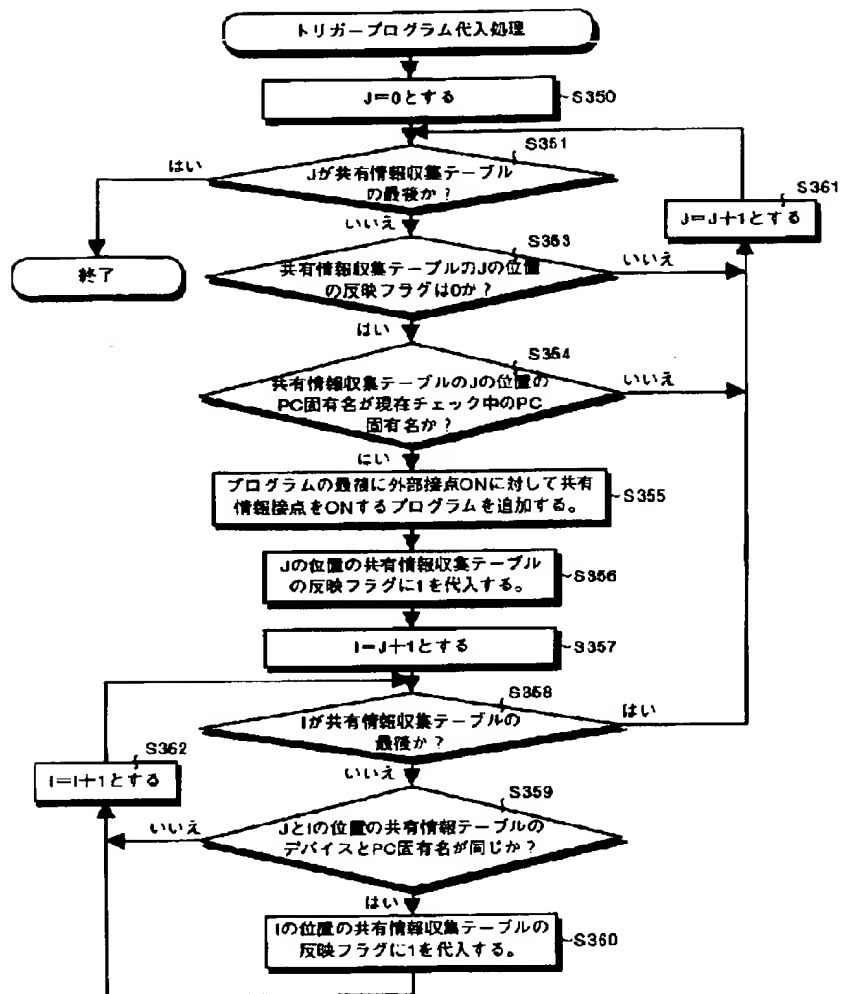
【図11】



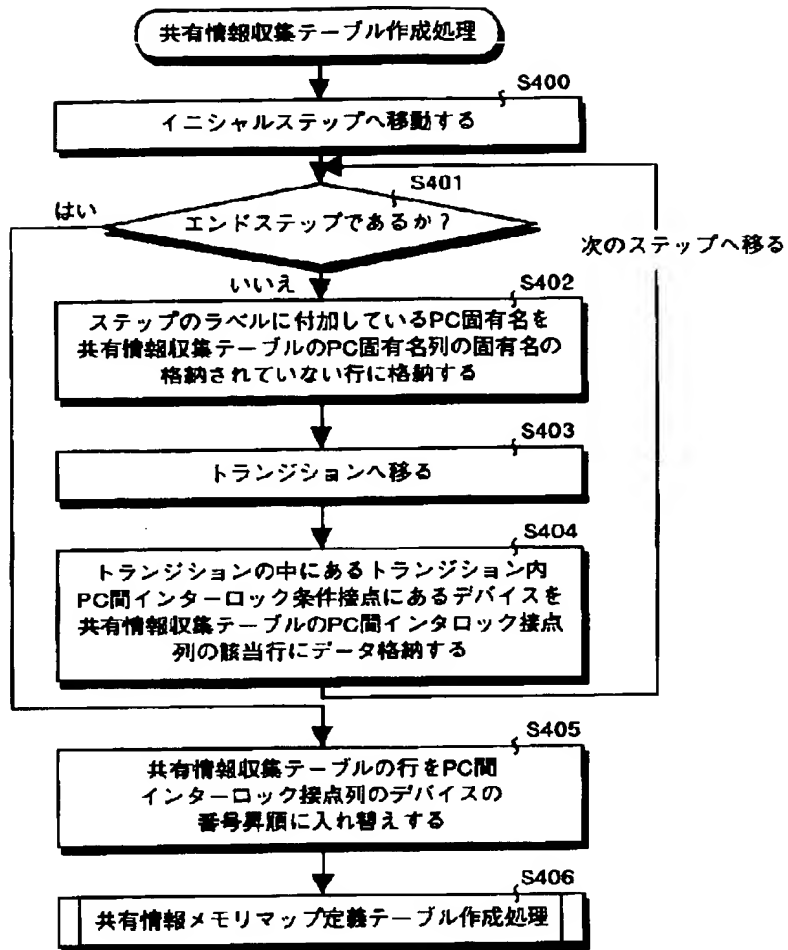
【図12】



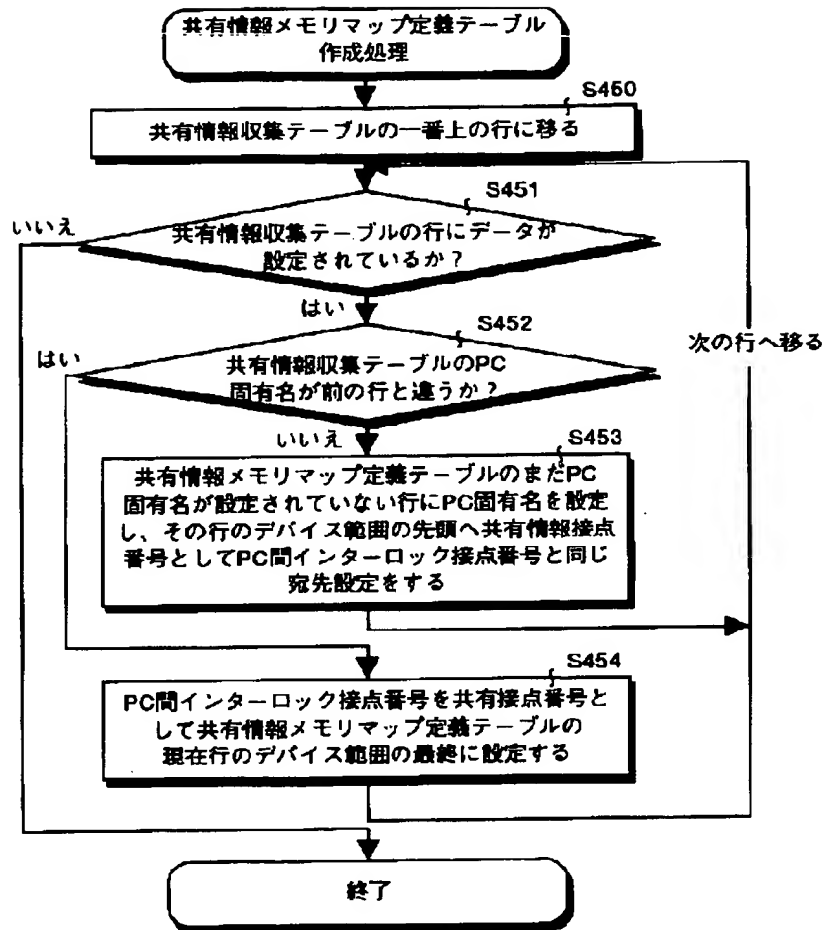
【図13】



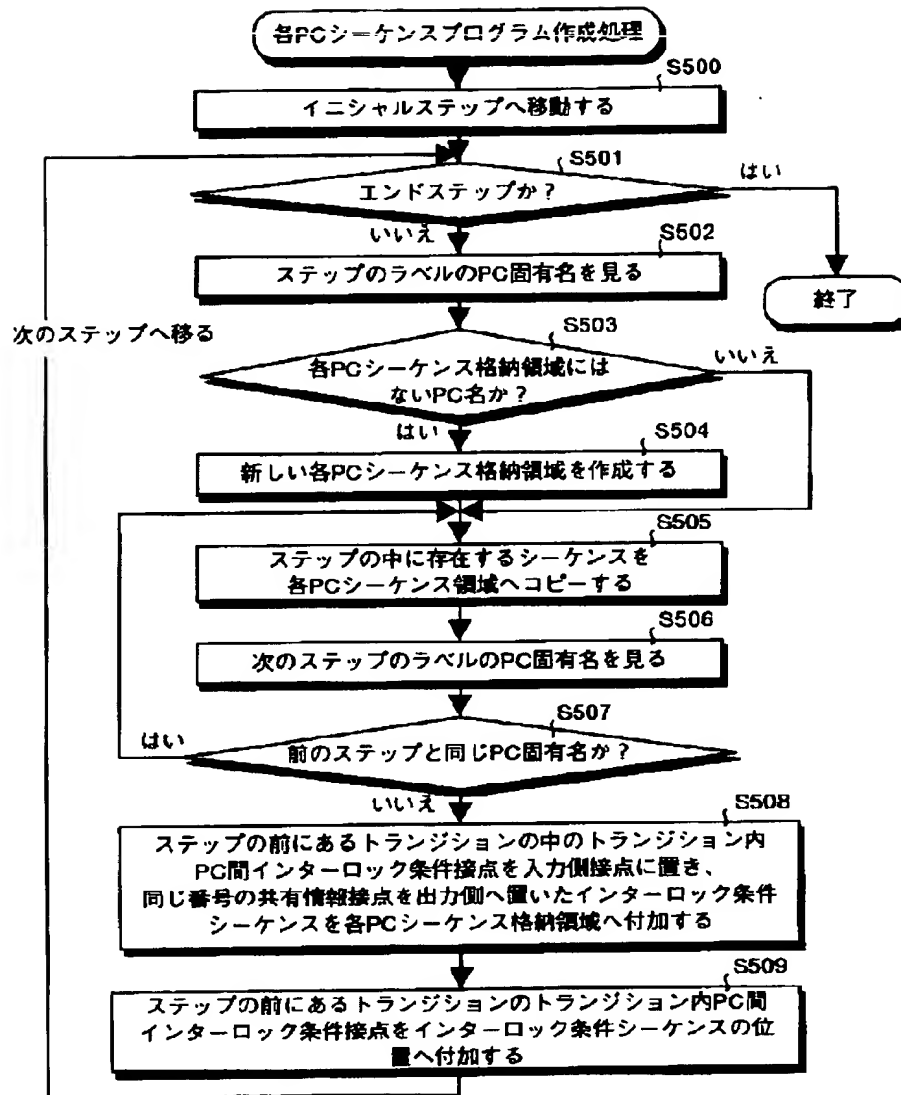
【図20】



【図21】



【図 22】



フロントページの続き

(72)発明者 山田 茂樹  
愛知県名古屋市中北区東大曽根町上五丁目  
1071番地三菱電機メカトロニクスソフトウ  
エア株式会社内

(72)発明者 落合 香月  
愛知県名古屋市中北区東大曽根町上五丁目  
1071番地三菱電機メカトロニクスソフトウ  
エア株式会社内

(72)発明者 大口 修司  
愛知県名古屋市中北区東大曽根町上五丁目  
1071番地三菱電機メカトロニクスソフトウ  
エア株式会社内

Fターム(参考) 5H220 AA04 BB12 CC07 CC09 CX02  
DD04 EE08 FF05 GG03 GG14  
HH04 JJ12 JJ26 JJ42 JJ59